CLOUD CV DIGITAL AMPLIFIERS

Serial Control Protocol

CLOUD ELECTRONICS LIMITED

© Cloud Electronics Limited
140 Staniforth Road,
Sheffield S9 3HF,
England
Phone +44 (0)114 244 7051.
Fax +44 (0)114 242 5462

E-mail technical@cloud.co.uk

Website www.cloud.co.uk

INTRODUCTION

The RS232 Serial Interface and the Ethernet Interface provide serial control of almost every CV Amplifier function, parameter and setting.
The RS232 interface is set to 8 data bits, one stop bit, no parity; default baud rate is 9600.
Default Ethernet port is 4999.

Throughout this document all message text is shown in a `mono spaced font`.

CV Amplifier MODELS

The CV range has three models having two, four, and eight amplified channels. Each amplifier has two auxiliary line level output channels. There is an option for an interface card which will give eight auxiliary inputs.

Model	Line Inputs	Auxiliary Inputs (Optional)	Amplifier Channels	Auxiliary Channels	Total Channels
CV8125	8	8	8	2	10
CV4250	4	8	4	2	6
CV2500	2	8	2	2	4

The examples in this document are targeted at a CV8125.

Blocks

Commands are sent to a block of channels.
A block is a set of channels that have been combined to be controlled together. EG if two channels are joined to form a stereo block, the command to change level is sent to the block rather than the individual channels.
If a channel is required to work independently it will be set as a mono block.

The factory default block configuration is for each channel to be set as a mono block.

Block Configuration.

Block configuration allows channels to be arranged into working blocks.
The block types are mono, stereo, two way mono crossover, two way stereo crossover with one low pass channel, and two way stereo crossover with two mono low pass channels.

Commands are available to enable and disable changing the configuration, query the amplifiers configuration, query the configuration of a single block, and set a new configuration.

Procedure

Block configuration has to be enabled before any changes can be made. The command `<BL.CF,E/>` sets the enabled state. The command `<BL.CF,D/>` will turn off the enabled state. The enabled state times out after 60 seconds.
The configuration is changed by the Configuration Set command. This command sends a list of tokens for the required blocks.

Each DSP in the CV amplifier controls two amplifier channels. If stereo block is required the Left and Right channels must be in the same DSP to allow their limiters to link. The tokens to define blocks also define how any channels not required by the block are used within the DSP. EG if a mono block was required in the first channel of a DSP the second channel would also have to be set as a mono block as it is not available to pair with another channel. Therefor the token for a mono block sets two mono blocks. A stereo crossover requires three channels so uses up two DSPs with a channel spare. There are two tokens for setting a stereo crossover. One token will set the spare channel of the DSP as a mono block; another token will set it as a second low pass channel of the crossover.

The tokens are:-
- "DM" Dual Mono. Set two channels within a DSP to two mono blocks.
- "ST" Stereo. Set the pair of channels within a DSP to a stereo block.
- "2WMM" Two way mono crossover. Set the pair of channels in a DSP so that the odd channel is the crossover high pass and the even channel is the crossover low pass.
- "2WSM" Two way Stereo crossover with a spare mono. Set the pair of channels within a DSP to a stereo crossover high pass; the crossover low pass is on the odd channel of a second DSP. The second DSP has a spare channel which is set as a Mono block.
- "2WSS" Two way stereo crossover with two low pass channels. Set the pair of channels within a DSP to a stereo crossover high pass; the crossover low pass is on both channels of a second DSP.

The block specified by the first token in the list starts at channel one; channels are then assigned in numerical order to the blocks in the list order.

If there are insufficient tokens for the available channels the remaining channels are set as mono blocks.

NB in an eight channel amplifier channels 9 and 10 are aux channels.

Examples, responses are shown in red:-
- `<BL.CF,E/>``<bl.cf,e/>`
  `<BL.CF,C=DM,ST,ST/>``<bl.cf,c=mo,mo,st,st,mo,mo,mo,mo/>`
  Block 1 has channel 1 as a mono block, block 2 has channel 2 as a mono block, block 3 has channels 3 and 4 as a stereo block, and block 4 has channels 5 and 6 as a stereo block. Channels 7, 8, 9 and 10 are assigned as mono blocks 5, 6, 7, and 8.
- `<BL.CF,E/>``<bl.cf,e/>`
  `<BL.CF,C=2WM,2WSM,2WSS/>``<bl.cf,c=2mo,2st,mo,2ss/>`
   Block 1 has channels 1 and 2 as a mono crossover block, block 2 has channels 3, 4, and 5 as a stereo crossover block, block 3 has channel 6 as mono block, and block 4 has channels 7, 8, 9, and 10 as a stereo crossover with two low pass channels

Delay assignment.

NB changing the block configuration deletes any assigned delays. You must configure the blocks before the assigning the delay.

The assigned delay is the set of current active delays in the amplifier. The delay build is used to set up a new delay assignment; it is an image of the required delays. When all the required delays have been added to the build it is copied to the assigned delay and the new delays from the build become active.

Commands are available to query the assigned delay, clear the assigned delay, modify the assigned delay, start a new build, add to a build, and query available space in the build.

The available delay is made up of multiple buffers. There are seven 100ms, two 50ms, and one 650ms buffer; all are tapped at 1ms intervals. A buffer can only be assigned to a single channel. The delay assigned to a channel will take one or more buffers. The delay manager will try to allocate the delay buffers in the most efficient way. Note that a stereo block requires twice the delay.

Build procedure

A new empty build is started by sending the Delay New command to the Delay Build sub destination. New delays for blocks are added to the build by sending the Delay Time Milliseconds command to the Delay Build sub destination of a block. When all required delays are added the build is copied to the assigned delay be sending the Delay Set command to the Delay Build sub destination; the new delays from the build then become active.

It is possible to modify a block's assigned delay without creating a new build using the Delay Time Milliseconds command with the "Up" or "Down" modifiers. NB the UP command can only increase the delay by the amount of delay available in the buffer assigned to the delay. EG if a delay was set at 20ms and assigned to a 100ms buffer it is possible to increase the delay to 100ms.

MESSAGE STRUCTURE

There are three types of messages in the CV amplifier control protocol. They are:-
- Control messages.
- Response messages.
- Error messages.

Control messages are sent to the CV amplifier. Response and error messages are transmitted from the CV amplifier in reply to a control message.

All control messages use upper case letters, all response messages are in lower case with one exception; a response to a label query returns the string of the label which may have upper or lower case characters. Error messages may be a mixture of upper or lower case.

The message body is enclosed within a header character ("<") and a terminator sequence ("/>"). The CV amplifier will reset its message decoding software upon reception of a header character, discarding any previously un-decoded partial messages. The CV amplifier will start to decode a message upon reception of the terminator sequence. If the decoded message is valid it is executed and a response message returned. An error message is returned if the message cannot be decoded or if the command cannot be executed.

THE CONTROL MESSAGE

Control messages are sent to the CV amplifier in order to perform a function. The control message has two fields separated by a comma. They are called the destination field and the command field.

`<DESTINATION,COMMAND/>`

DESTINATION FIELD

The destination field is an abbreviated description of the section the command is intended to affect. The field consists of a two or three character **main destination** an optional two character **destination channel** and an optional two character **sub destination**. The destination channel is separated from the main destination by a colon. The sub destination is separated from the main destination, or destination channel, by a full stop (period).

MAIN DESTINATION

CV amplifiers have five possible main destinations;
1. Line inputs.
2. Auxiliary inputs.
3. Blocks.
4. Channel.
5. System.

DESTINATION CHANNEL

The destination channel is used to address a message to an individual channel in a block. There are two destination channels, `HF` for the High Frequency channel of a crossover and `LF` for the Low Frequency channel of a crossover.
Example
`<B1:HF.S3,B5>` HF is the destination channel for the High Frequency band of a crossover.

SUB DESTINATION

The sub destination is used to address the message to a particular part of a main destination or its destination channel. Block and System destinations have sub destinations.

Examples
- `<B1,S3/>` B1 is the main destination.
- `<B1.RE,E/>` B1 is the main destination, `RE` is the sub destination.
- `<B1:HF,IE/>` B1 is the main destination, `HF` is the destination channel.
- `<B1:HF.S3,B5/>` B1 is the main destination, `HF` is the destination channel, `S3` is the sub destination.

There are two types of destination and sub destination identifiers; **absolute** and **indexed**.

- **ABSOLUTE DESTINATION IDENTIFIER.**
  This is used when the destination has only one member such as the **System**, or when the destination has members but the message is for all members. EG **Block** is an absolute destination for configuring blocks. An absolute destination's identifier is two alphabetic characters.

- **INDEXED DESTINATION IDENTIFIER**.
  This is used when the destination has multiple members and the message is intended for an individual member. Blocks, channels, auxiliary inputs and line inputs are destinations which have multiple members. Indexing allows addressing members such as block 1 or block 2. An indexed destination identifier consists of an alphabetic character to identify the destination type and a numeric index for the member. Indexes out of range are not allowed, EG B12 and L0 are invalid.

Examples
- `<SY,IF/>` SY is an absolute destination identifier for System.
- `<B1,L23/>` B1 is an indexed destination identifier for Block1.

COMMAND FIELD

A command field identifies the command to be performed on the destination. The command field may have up to three parts; the **command identifier**, the **command modifier** and the **command value**. No white space is permitted between these parts.

- **COMMAND IDENTIFIER**.
  This part is always required. It is a single alphabetic character to identify the command type.
- **COMMAND MODIFIER.**
  This part is optional depending on the command type. Some commands take no modifier, some commands always take a modifier and some commands can have the option of a modifier or none. The modifier is a single alphabetic character.
- **COMMAND VALUE**.
  This part is optional depending on the command type and modifier. The value part can be any sequence of characters as required by the command type. In cases where a value can contain alphabetic characters it is preceded by an equals character ("=") to prevent the first character of the value being confused with a command modifier.

DESTINATIONS IN DETAIL

MAIN DESTINATIONS

AUXILIARY INPUT INDEXED
Auxiliary Input Indexed does not take a sub destination. Its use is to change auxiliary input labels. Auxiliary inputs are available from the optional Digital Interface Card. Valid main destination identifiers are: `A1, A2, A3, A4, A5, A6, A7,` and `A8`.

BLOCK
Block is the destination for commands to configure blocks.
The identifier is: `BL`.

BLOCK INDEXED
Block indexed is the destination for commands to a block of channels. A block is how an amplifier channel or group of channels are organised i.e. mono, stereo or crossover. An eight channel amplifier has a maximum of ten blocks, a four channel amplifier has a maximum of six blocks and a two channel amplifier has a maximum of four blocks. This is because all amplifier channels and the two auxiliary channels could be set up as mono blocks.
Valid identifiers are: `B1, B2, B3, B4, B5, B6, B7, B8, B9` and `B10`.

CHANNEL INDEXED
Channel Indexed does not take a sub destination. Its use is to get meter values from the individual amplifier channels. An eight channel amplifier has a maximum of ten Channel Indexed destinations, a four channel amplifier has a maximum of six Channel Indexed destinations and a two channel amplifier has a maximum of four Channel Indexed destinations. This is because the total of Channel Indexed destinations is the number of amplifier channels plus the two auxiliary channels. Valid main destination identifiers are: `C1, C2, C3, C4, C5, C6, C7, C8, C9` and `C10`.

LINE INPUT INDEXED
Line Input Indexed does not take a sub destination. Its use is to change line input labels. Valid main destination identifiers are: `L1, L2, L3, L4, L5, L6, L7,` and `L8`.

SYSTEM
System is used to set system wide parameters of the CV amplifier such as the start-up mode, text field, RS232 settings, delay, password, GPIO, bootloader mode, APD, RTC calibration, mute socket, query hardware and software versions, and to command a system reset.
The identifier is: `SY`.

DESTINATION CHANNELS

HF
This is for the high frequency channel of a crossover
The identifier is: `HF`.
Example `<B3:HF.EQ,L12/>`

LF
This is for the low frequency channel of a crossover.
The identifier is: `LF`.
Example `<B3:LF.S1,F100/>`

SUB DESTINATIONS

AUXILIARY EQ
This is a sub destination of a Block Indexed destination that is made up only of Auxiliary channels that is not set as a cross over. It is used to set and query the parameters of a crossover filter so that the block can behave as a crossover channel of another block.
Example identifiers are `B7.AE`, and `B10.AE`.

AUXILIARY LEVEL SLAVE
This is a sub destination of a Block Indexed destination that is made up only of Auxiliary channels. It is used to set and query the slaving of level to another block.
Example identifiers are `B7.AL`, and `B10.AL`.

AUXILIARY MUTE SLAVE
This is a sub destination of a Block Indexed destination that is made up only of Auxiliary channels. It is used to set and query the slaving of muting to another block.
Example identifiers are `B7.AM`, and `B10.AM`.

AUXILIARY SOURCE SLAVE
This is a sub destination of a Block Indexed destination that is made up only of Auxiliary channels. It is used to set and query the slaving of source selection to another block.
Example identifiers are `B7.AS`, and `B10.AS`.

CONFIGURATION
This sub destination is only available to the Block destination. It is used to query, enable, disable, and change block configuration.
The only valid main destination and sub destination identifier combination is `BL.CF`.

CROSSOVER HIGH
This is a sub destination of a Block Indexed destination that is assigned as a crossover. It is used to set and query the frequency and slope of the high pass channel of the crossover.
Example identifiers are `B1.XH`, and `B3.XH`.

CROSSOVER LOW
This is a sub destination of a Block Indexed destination that is assigned as a crossover. It is used to set and query the frequency and slope of the low pass channel of the crossover.
Example identifiers are `B6.XL`, and `B2.XL`.

DELAY
This is a sub destination of a Block Indexed destination or the System destination. It is used to reset and query the delay assigned to the system and to query and increment or decrement the delay assigned to a block.
Example identifiers are `B1.DY`, and `SY.DY`.

DELAY BUILD
This is a sub destination of a Block Indexed destination or the System destination. It is used to start, cancel, query and assign a delay build to the system and to reserve delay or query reserved delay in block.
Example identifiers are `B1.DB`, and `SY.DB`.

EQ BALANCE
This is a sub destination of a Block Indexed destination that is assigned as a crossover. It is used to set and query the speaker level trim. It also requires a destination channel.
Example identifiers are `B1:LF.EQ`, and `B3:HF.EQ`.

GPIO
This is a sub destination of the System destination. It is used to set and query the modes and states of the General Purpose Input Output.
Valid identifiers are `SY.G1, SY.G2` and `SY.G3`.

HARDWARE VERSION `"HV"`
SOFTWARE VERSION `"SV"`
These are sub destinations of the System destination. They are used to identify the CV amplifier's versions for hardware and software.
Example identifiers are `SY.HV,` and `SY.SV`.

INPUT METER
This sub destination is only available to the Channel Indexed destination. It is used to query the input meter of a channel.
Valid main destination and sub destination identifier combinations are
`C1.MI, C2.MI, C3.MI, C4.MI, C5.MI, C6.MI, C7.MI, C8.MI, C9.MI,` and `C10.MI`.

LIMITER
This sub destination is only available to the Block Indexed destinations. It allows setting and querying limiter parameters such as threshold and limiter enable, and querying the limiter's Gain Reduction.
Valid main destination and sub destination identifier combinations are
`B1.LI, B2.LI, B3.LI, B4.LI, B5.LI, B6.LI, B7.LI, B8.LI, B9.LI,` and `B10.LI`.

OUTPUT METER
This sub destination is only available to the Channel Indexed destinations. It is used to query the output meter of a channel.
Valid main destination and sub destination identifier combinations are `C1.MO`, `C2.MO`, `C3.MO`, `C4.MO`, `C5.MO`, `C6.MO`, `C7.MO`, `C8.MO`, `C9.MO`, and `C10.MO`.

ROOM EQ
This is a sub destination of a Block Indexed destination. It is used to set and query the bypass state of the seven band room EQ.
Example identifiers are `B1.RE`, and `B3.RE`.

ROOM EQ BASS
This is a sub destination of a Block Indexed destination. It is used to set and query Cut/ Boost parameters of band 1 in the room EQ when band 1 is used as a bass filter.
Example identifiers are `B1.RB`, and `B3.RB`.

ROOM EQ HIGH SHELF
This is a sub destination of a Block Indexed destination. It is used to set and query frequency and Cut/ Boost parameters of band 7 in the room EQ when band 7 is used as a high shelf filter.
Example identifiers are `B1.RH`, and `B3.RH`.

ROOM EQ INDEXED
This is a sub destination of a Block Indexed destination. It is used to set and query the frequency, Q and Cut/ Boost parameters of one of the seven bands in the room EQ.
Example identifiers are `B1.R5`, and `B3.R7`.

ROOM EQ LOW SHELF
This is a sub destination of a Block Indexed destination. It is used to set and query frequency and Cut/ Boost parameters of band 1 in the room EQ when band 1 is used as a low shelf filter.
Example identifiers are `B1.RL`, and `B3.RL`.

ROOM EQ TREBLE
This is a sub destination of a Block Indexed destination. It is used to set and query Cut/ Boost parameters of band 7 in the room EQ when band 7 is used as a treble filter.
Example identifiers are `B1.RT`, and `B3.RT`.

RS232

This is a sub destination of the System destination. It is used to set and query the RS232 Baud rate.

The only valid main destination and sub destination identifier combination is `SY.RS`.

RTC

This is a sub destination of the System destination. It is used to set and query the Real Time Clock calibration.

The only valid main destination and sub destination identifier combination is `SY.RC`.

SPEAKER DELAY

This is a sub destination of a Block Indexed destination that is assigned as a crossover. It is used to set and query the speaker alignment delay. It also requires a destination channel.

Example identifiers are `B1:LF.SD`, and `B3:HF.SD`.

SPEAKER EQ

This is a sub destination of a Block Indexed destination. It is used to set and query the bypass state of the five band speaker EQ. A destination channel is required when used a sub destination of a block set up as crossover.

Example identifiers are `B1.SE`, and `B3:HF.SE`.

SPEAKER EQ INDEXED

This is a sub destination of a Block Indexed destination. It is used to set and query the frequency, Q and Cut/ Boost parameters of one of the five bands in the speaker EQ. A destination channel is required when used a sub destination of a block set up as crossover.

Example identifiers are `B1.S5`, and `B3:HF.S2`.

SPEAKER PROTECT HIGH PASS FILTER

This is a sub destination of a Block Indexed destination. It is used to set and query the speaker protect high pass filter. A destination channel is required when used a sub destination of a block set up as crossover.

Example identifiers are `B1.PF`, and `B3:HF.PF`.

TEXT FIELD

This is a sub destination of the System destination. It allows a 32 character ASCII label, free for any use. EG a description of the unit can be used to identify between units on a network.

Example identifier is `SY.TX`.

COMMANDS IN DETAIL

Please see the examples section.

BAUD "B"

This command is only available to the RS232 sub destination of the System destination. It takes a numeric value which must be one of the available baud rates on the CV amplifier. The response is a lower case echo of the command sent. This command should be used with caution over RS232 since it will change the baud rate after the response has been transmitted.

Available baud rates are 4800, 9600, 19200, 38400, 57600, and 115200.

The command options are:-

- "Bn" for **Baud.**
  The value n is the Baud rate.
- "BQ" for **Query.**

BOOT LOAD MODE "B"

This command is for the System destination. The command is used to access the Bootloader mode of the CV amplifier.

When bootloader is enabled the CV amplifier will enter the bootloader at next power on or when reset. When disabled the CV amplifier will enter the application mode (normal running) at power on or reset.

The command must have a modifier. The modifier can be used to lock or unlock modifications to the bootloader mode or to enable, disable or query the bootloader mode.

As a safety feature enabling or disabling is only available when unlocked. As an extra safeguard the unlock modifier requires the Password Key as a value.

The command options are:-

- "BD" for **Disable.**
- "BE" for **Enable.**
  Boot load is enabled. If the Boot Load Reset command is now sent or the power cycled the amplifier will restart in bootloader mode.
  NB To exit Bootloader mode either load new firmware or power cycle again.
- "BL" for **Lock.**
  Prevent boot load from being enabled. If boot load was enabled it is disabled.
- "BUnnnn" for **Unlock.**
  The value nnnn is the 4 digit password key.
- "BQ" for **Query.**
- "BR" for **Reset.**
  Reset the amplifier in boot load mode. This will return error if boot load is not enabled.

CONFIGURATION SET "C="
This command is sent to the Block destination to set how the blocks are configured. It will only work if Block Configuration has been enabled. The value field is a comma separated list of tokens that define the configuration in order from block 1 onwards. If there are insufficient tokens, unassigned channels are set as Mono Blocks.
NB changing the block configuration deletes any assigned delays.
The tokens are:-

- "DM" Dual Mono. Two channels set as two Mono blocks.
- "ST" Stereo. Two channels set as one Stereo block.
- "2WM" Two way Mono crossover. Two channels as one crossover block.
- "2WSM" Two way Stereo crossover with a spare Mono block. Three channels as a crossover block with Stereo HF and a Mono LF. One channel as a Mono block.
- "2WSS" Two way stereo crossover with additional LF. Four channels as a crossover block with stereo HF and two Mono LF channels.

CROSSOVER OVERLAP DISABLE "OD"
CROSSOVER OVERLAP ENABLE "OE"
These commands are sent to the Crossover High or Crossover Low sub destinations of a Block Indexed destination that is assigned as a crossover. Overlap is disabled by default. When overlap is disabled the frequency of Crossover High channel is the same as Crossover Low channel; a command to change the frequency of one changes the frequency of the other. When overlap is enabled Crossover High channel and Crossover Low channel can have different frequencies; a command to change the frequency of one channel does not change the frequency of the other.

CROSSOVER SLOPE BESSEL "BE"
CROSSOVER SLOPE BUTTERWORTH "BU"
CROSSOVER SLOPE LINKWITZ RILEY "LR"
These commands are sent to the Crossover High or Crossover Low sub destinations of a Block Indexed destination that is assigned as a crossover. They can also be sent to the Auxiliary EQ sub destination of a Block Indexed destination not assigned as a crossover if the block only contains auxiliary channels. The command takes a numeric value for the dBs per octave of the slope. Permitted values are ("12"), ("18") and ("24") for Bessel and Butterworth, and ("12"), ("24"), ("36") and ("48") for Linkwitz Riley.
Example commands are:-

- "BE12" Bessel 12 dBs per octave.
- "BU24" Butterworth 24 dBs per octave.
- "LR35" Linkwitz Riley 36 dB per octave.

DELAY AVAILABLE QUERY `"AQ"`

This command is sent to the Delay Build sub destination of the System destination. It returns the delay time in milliseconds available in the build. IE the delay time that has not been reserved.

DELAY NEW `"N"`

This command is sent to the Delay Build sub destination of the System destination. It starts a new empty Delay Build.

NB Block configuration deletes the current delay assignment. Do not assign delays until you have configured the blocks of channels.

DELAY RESET `"R"`

This command can be sent to the Delay sub destination or the Delay Build sub destination of the System destination. When sent to the Delay sub destination it clears any assigned delays. When sent to the Delay Build sub destination it clears any delays reserved in the Delay Build.

DELAY SET `"S"`

This command is sent to the Delay Build sub destination of the System destination. It assigns the delay build to be the active delay.

NB Block configuration deletes the current delay assignment. Do not assign delays until you have configured the blocks of channels.

DELAY TIME MILLISECONDS **"M"**

This command without a modifier is sent to the Delay Build sub destination of a Block Indexed destination to reserve delay for the block in the build. The value field is the delay time in milliseconds. The maximum value is dependent on the delay available in the build; as more delay is reserved less becomes available. NB a stereo type block will reserve twice as much delay in the build as Left and Right will require separate delays.

The command options are:-

- **"Mn"** for **Reserve Millisecond.**
  The value n is the time in milliseconds to reserve.
- **"MD"** for **Down.**
  This command is sent to the Delay sub destination of a Block Indexed destination. It allows decrementing of the block's assigned delay.
- **"MU"** for **Up.**
- This command is sent to the Delay sub destination of a Block Indexed destination. It allows limited incrementing of the block's assigned delay.
- **"MQ"** for **Query.**
  Sent to the Delay Build sub destination of a Block Indexed destination it returns the delay reserved for the block in the build.
  Sent to the Delay sub destination of a Block Indexed destination it returns the delay assigned to the block.
  Sent to the Delay sub destination of a Block Indexed destination it returns the delay assigned to the block.
  Sent to the Delay sub destination of the System destination it returns the total delay assigned.

DELAY TIME MICROSECONDS **"U"**

This command performs speaker alignment delay between channels of a crossover. It is sent to the Speaker Delay sub destination of a Block Indexed destination. It is only available to blocks set as a crossover and requires a destination channel.

It takes a numeric value between 1 and 30000 micro seconds. The Query modifier ("Q") returns the delay value.

NB Block configuration deletes the current speaker delay assignment. Do not assign speaker delays until you have configured the blocks of channels.

The commands are:-

- **"Uxxxx"** for **Delay Time Microseconds.**
  The value xxxx is the time from 0 to 30000 microseconds.
- **"Uxxxx"** for **Delay Time Microseconds Query.**
  Returns the delay time set on the speaker in microseconds.

DISABLE `"D"`
ENABLE `"E"`
These two commands are used to enable/ disable some sub destinations of the Block or the Block Indexed destinations. The commands take no modifiers or values. To query a sub destination's enable/ disable state issue the Query command.
Valid sub destinations are:-

- The Configuration sub destination of a Block destination.
  Enable/ disable configuration of blocks.
- The Room EQ sub destination of a Block Indexed destination.
  Enable/ disable all bands of the Room EQ.
- The Speaker Protect High Pass Filter sub destination of a Block Indexed destination.
  Enable/ Disable the Speaker protect High Pass Filter.
- The Speaker EQ sub destination of a Block Indexed destination.
  Enable/ disable all bands of the Speaker EQ.
- The Limiter sub destination of a Block Indexed destination.
  Enable/ disable the limiter.
- The Disable command can be sent to the Auxiliary EQ sub destination of a Block Indexed destination that is not set as a crossover and contains only Auxiliary channels. NB the Auxiliary EQ sub destination is enabled by sending an EQ High Pass or EQ Low Pass command.

EQ BALANCE `"L"`
This command is sent to the Crossover High or Crossover Low sub destinations of a Block Indexed destination that is assigned as a crossover to adjust the level in the crossover channel. It requires a Destination channel.
The command options are:-
- `"Ln"` for **Balance Level.**
  The value n is the balance level.
- `"LQ"` for **Query.**

EQ BOOST `"B"`
EQ CUT `"C"`
These commands send a value to set the cut or boost of a filter. The value is in dB from 0 to 12.
The Boost and Cut commands can be sent to the following sub destinations:-

- The Room EQ High Shelf or the Room EQ Low Shelf sub destinations of a Block Indexed destination.
- The Room EQ Indexed sub destination of a Block Indexed destination.
- The Room EQ Bass or the Room EQ Treble sub destinations of a Block Indexed destination.
- The Speaker EQ Indexed sub destination of a Block Indexed destination. A Destination Channel is required if the block is set as a crossover.

EQ FREQUENCY "F"

This command sends a value to set the frequency of a filter. The value is in Hertz from 10 to 20000.

The Frequency command can be sent to the following sub destinations:-

- The Room EQ High Shelf or the Room EQ Low Shelf sub destinations of a Block Indexed destination.
- The Room EQ Indexed sub destination of a Block Indexed destination.
- The Crossover High or the Crossover Low sub destinations of a Block Indexed destination set as a crossover.
- The Auxiliary EQ sub destination of a Block Indexed destination that is not set as a crossover and contains only Auxiliary channels.
- The Speaker Protect High Pass Filter sub destination of a Block Indexed destination. A Destination Channel is required if the block is set as a crossover.
- The Speaker EQ Indexed sub destination of a Block Indexed destination. A Destination Channel is required if the block is set as a crossover.

EQ HIGH PASS "HP"
EQ LOW PASS "LP"

These commands are sent to the Auxiliary EQ sub destination of a Block Indexed destination that is not set as a crossover and contains only Auxiliary channels. They set the Auxiliary EQ sub destination to high pass or low pass.

EQ Q "Q"

This command is sent to the Room EQ Indexed or Speaker EQ Indexed sub destination of a Block Indexed destination. Its value sets the sub destinations Q. It requires a Destination Channel when sent to the Speaker EQ Indexed sub destination of a block set as a crossover.

An example command is "Q1.7".

GPIO MODE MUTE IN "MI"
GPIO MODE FAULT OUT "FO"
GPIO MODE QUERY "MQ"
GPIO MODE USER IN "UI"
GPIO MODE USER OUT "UO"

These commands set and query the mode of each GPIO terminal. They are sent to the GPIO sub destination of the System destination.

The GPIO Mode commands are:-

- "MI" **Mute In** sets the GPIO terminal as an input which will mute the CV amplifier when asserted.
- "FO" **Fault Out** sets the GPIO terminal as an output which signals a fault condition in the CV amplifier.
- "MQ" **Query** returns the mode of the GPIO terminal.
- "UI" **User In** sets the GPIO terminal as a user input. The state can be examined with the **GPIO State Query** command.
- "UO" **USER Out** sets the GPIO terminal as an output which can be set or cleared using the **GPIO State Set** and **GPIO State Clear** commands.

GPIO STATE CLEAR "C"
GPIO STATE SET "S"
GPIO STATE QUERY "SQ"
These commands set and query the state of each GPIO terminal. They are sent to the GPIO sub destination of the System destination.
The GPIO State commands are:-

- **"C"** **Clear.**
  Clear the state of a GPIO terminal that is in the User Output Mode.
- **"S"** **Set.**
  Set the state of a GPIO terminal that is in the User Output Mode.
- **"SQ"** **State Query.**
  Request the state of a GPIO terminal set to any mode.

INITIALISATION "I"
This command is sent to the System destination. The command is used to set the Initialisation mode of the CV amplifier i.e. the settings at start up. The command must take a modifier.
The commands options are:-

- **"IF"** for **Factory**.
  Restore to factory settings. NB the factory setting for Initialisation is Previous Mode.
- **"IP"** for **Previous**.
  The amplifier powers up exactly as it was at power down.
- **"IQ"** for **Query**.

INVERT "I"
This command is used to set the phase of an output in a Block Indexed destination. A Destination Channel is required If the block is assigned as a crossover. The command must take a modifier.
The command options are:-

- **"IE"** for **Enable**.
  Phase is inverted.
- **"ID"** for **Disable**.
  Phase is normal.
- **"IQ"** for **Query**.
  Query the phase.

KEY "K"

This command is sent to the System destination. It has no modifiers.
It is used to change the four digit numeric password. The value field has eight numeric characters. The first four characters are the old password; the second four are the new password.

LEVEL "L"

This command is used to set the level in a Block Indexed destination.
An absolute value is the attenuation required in one dB steps so an absolute value of 20 represents 20 dB attenuation or -20dB. The up and down modifiers also require a value for the change in level.
The minimum attenuation is 0dB, the maximum attenuation is 90dB.
The command options are:-

- "Ln" for **Absolute Level.**
  The value n is the absolute value of level to set; 0dB to -90dB.
- "LUn" for **Up.**
  The value n is the dB change in level.
  EG A value of 10 will increase the level by 10dB.
- "LDn" for **Down.**
  The value n is the dB change in level.
  EG A value of 10 will decrease the level by 10dB.
- "LQ" for **Query.**
  Return the current value of level and parser control enable state.
- "LX" for **Disable.**
  Disable parser control over level. Obey rear level control.
- "LE" for **Enable**.
  Enable parser control over level. Ignore rear level control.

The response message to any adjustment (up or down) when sent to a Block Indexed destination is a lower case version of an absolute level command.
NB Blocks containing only Auxiliary channels will ignore the Absolute, up and down commands if Source and Level Slaving are enabled. Auxiliary channels have no rear level control so Disable/ Enable return Unavailable error when applied to blocks containing only Auxiliary channels.

LIMITER GR QUERY "GQ"

This command is sent to the Limiter sub destination of a Block Indexed or a Channel Indexed destination. It returns the dB value of the limiter's Gain Reduction.

LIMITER THRESHOLD "T"

This command is sent to the Limiter sub destination of a Block Indexed destination. It takes a value which sets the Limiter Threshold in dBs below 0dB.

LINE VOLTAGE "V"

This command is sent to a Block Indexed destination to set the Output Line Voltage to 100V or 70V.
The command options are:-

- "V70" for **70V**.
- "V100" for **100V**.
- "VQ" for **Query**.
  Returns the voltage and parser control enable state.
- "VE" for **Enable**.
  Enable parser control. Ignore the rear panel DIP switch.
- "VD" for **Disable**.
  Disable parser control. Enable rear DIP switch control.

NB. Auxiliary channels have no output amplifiers so these commands will return Unavailable error when applied to blocks containing only auxiliary channels. destination.

MUTE "M"
OPEN "O"

These commands set ("M") and clear ("O") the Mute Status. They are sent to a Block Indexed destination. The command takes no modifier or value. The response message is a lower case version of the control message. Blocks containing only Auxiliary channels will ignore these commands if Source and Mute Slaving are enabled.
NB to get the mute status use the query command sent to a Block Indexed destination.

NAME LABEL "N"

This command is used to set the name label on Line Input Indexed, Auxiliary Input Indexed and Block Indexed destinations. The command can take the query modifier. The unmodified command is followed by an equals character ("=") and an alphanumeric sequence. The alphanumeric sequence can contain any sequence of characters except "/>" which is interpreted as the terminator. Only the first sixteen characters are used to generate the label. The response message of this command is a lower case copy of the control message before the value followed by an equals character ("=") and then the character sequence set as the label.
The commands are:-

- "N=Lounge" for **Set Name**.
  The value "Lounge" is the label to set.
- "NQ" for **Query**.

PING "?"

This command is sent to the System destination. It has no modifiers.
It has no effect other than to trigger a response message.

POWER `"P"`
This command is sent to the System destination. It always takes a modifier.
The command options are:-

- `"PU"` for **Power Up**.
  Restore the CV amplifier from Power Down mode.
- `"PD"` for **Power Down**.
  Put the CV amplifier into Power Down mode.
- `"PQ"` for **Power Query**.
  Query the state of Power Down mode and the state of the REMOTE PWR
  DOWN socket. Returns a comma separated list. The first returned field can
  be `"pe"` for power enabled, and `"pd"` for power disabled. The second field
  can be `", e"` for socket not shorted and, `", d"` for socket shorted.

PARSER CONTROL ENABLE `"PE"`
PARSER CONTROL DISABLE `"PX"`
These commands are sent to the Speaker Protect High Pass Filter sub destination
of a Block Indexed destination. A Destination Channel is required if the block is set
as a crossover.
The commands are:-
- `"PE"` for **Parser Control Enable**.
  Enable parser control. Ignore the rear panel DIP switch.
- `"PX"` for **Parser Control Disable**.
  Disable parser control. Enable the rear panel DIP switch.

QUERY `"Q"`
This command makes a general query for additional information that is not
returned by any commands with a query modifier. Note that for many commands
have query available as a command modifier.
The Query command can be sent to the following destinations and sub
destinations:-

- The Configuration sub destination of a Block Indexed destination. Requests
  the block's' configuration.
- The Configuration sub destination of a Block destination. A comma
  separated list of all blocks' configurations is returned.
- The Crossover High or Crossover Low sub destination of a Block Indexed
  destination. A comma separated list of the Crossover's parameters is
  returned.
- The Speaker Protect High Pass Filter of a Block Indexed destination. A
  comma separated list of the EQ's parameters is returned.
- The Auxiliary EQ sub destination of a Block Indexed destination that
  contains only Auxiliary channels. A comma separated list of the EQ's
  parameters is returned.
- The Speaker or Room EQ sub destination of a Block Indexed destination.
  Requests the EQ's enable state.
- The Speaker or Room EQ band sub destination of a Block Indexed
  destination. A comma separated list of the band's parameters is returned.

- The Room EQ Bass, Room EQ Treble, Room EQ High Shelf or Room EQ Low Shelf sub destination of a Block Indexed destination. A comma separated list of the band's parameters is returned.
- The Limiter sub destination of a Block Indexed destination. A comma separated list of the threshold, and enable state is returned.
- The Meter sub destination of a Channel Indexed destination. The dB value of the meter is returned.
- The Block Indexed destination. The Mute state is returned; blocks containing only Auxiliary channels will return the master's mute state if source and mute slaving are enabled.
- The Text Field sub destination of the System destination. The text within the Text Field is returned.
- The Software Version or Hardware Version sub destination of the System destination. A text value for the version is returned.
- The RTC sub destination of the System destination. The RTC calibration value is returned.

RESET "R"

This command is sent to the System destination. It has no modifiers and takes no value. The command causes the CV amplifier to reset and restore factory settings.

RTC CALIBRATION STORE "CS"

This command is sent to the RTC sub destination of the System Destination. It is used to save the calibration value to NVM.

RTC CALIBRATION DECREMENT "CD"
RTC CALIBRATION INCREMENT "CU"

These commands are sent to the RTC sub destination of the System Destination. These commands send a numeric value which, when RTC calibration is enabled, is used to increment or decrement the calibration value. The maximum value that can be sent is 255.

RTC CALIBRATION DISABLE "CD"
RTC CALIBRATION ENABLE "CE"

These commands are sent to the RTC sub destination of the System Destination. They are used to Enable and Disable the RTC calibration mode. In calibration mode the RTC puts out a 1Hz signal on the test terminal marked RTCC_MFP.

SET TEXT "S="

This command is sent to the Text Field sub destination of the System destination. Its value field is up to 32 characters of text which are to be stored in the amplifier's Text Field. The text can be read back by sending the Query command to the Text Field sub destination of the System destination.

SLAVE AUXILIARY "s"

These commands are only sent to blocks that contain only Auxiliary channels; they allow the Source, Level and Mute functions to be slaved to another block. Auxiliary Mute and Auxiliary Level slaving only work when Auxiliary Source slaving is enabled.

The form of the command that has a value can only be sent to the Auxiliary Source Slave sub destination. It sets the index of the block that is to be the master.

Auxiliary Level Slave and Auxiliary Mute Slave sub destinations can be set to be enabled when Auxiliary Source Slave is disabled but the enabling does not take effect until Auxiliary Source Slave is enabled.

The command options are:-

- "Sn" for **Set Master.**
  The value n is the index of the master block.
- "SD" for **Disable.**
  Disable slaving of the sub destination.
- "SE" for **Enable.**
  Enable slaving of the sub destination. This command will have no effect on Auxiliary Level Slave and Auxiliary Mute Slave sub destinations if slaving is disabled in the Auxiliary Source Slave sub destination.
- "SQ" for **Query**.
  Request the slaving enable state of a sub destination. The Auxiliary Source Slave sub destination will also return the index of the block that it is slaved to.

SOURCE "s"

The command is used to set the line input selection in a Block Indexed destination. The source command requires either a modifier or an absolute value. An absolute value is the line input number from "1" to "8" or "0" for no line input selected. The command options are:-

- "Sn" for **Source.**
  The value n is the selected source.
- "SS" for **Stereo Sum.**
  Set source to sum stereo to mono. Only available to Mono and Mono Crossover blocks.
- "SM" for **Mono.**
  Set source to mono. This turns off source sum stereo to mono. Only available to Mono and Mono crossover blocks.
- "SA" for **Auxiliary inputs.**
  Use the source from the Auxiliary Inputs.
- "SL" for **Line inputs.**
  Use the source from the Line Inputs.
- "SQ" for **Query.**
  Returns a comma separated list of the source, the stereo summed source, input type (Auxiliary or Line) and the parser control enabled state.
- "SX" for **Disable.**
  Disable parser control over Input Source. Enable rear DIP switch control.

- "SE" for **Enable**.
   Enable parser control over Input Source. Ignore rear DIP switch control.

NB Blocks containing only Auxiliary channels ignore the absolute, auxiliary, line, stereo sum and mono commands if source and level slaving are enabled. Auxiliary channels have no rear level DIP switches so Disable/ Enable return Unavailable error when applied to blocks containing only Auxiliary channels.

WAKE UP "WU"
This command is for the System destination. It wakes the CV amplifier from Automatic Power Down.

THE RESPONSE MESSAGE

The response message is the CV amplifier's reply to a valid message. The message body contains a lower case version of the destination field and the command identifier followed by one or more values.

The responses in these examples are shown in red.

- `<B1,L12/>``<b1,l12/>`   Set the level in Block 1 to 12. Response returns the new value of level in Block 1.
- `<B1,LU3/>``<b1,l9/>`   Increment the level in Block 1 by 3. Response returns the new value of level in Block 1.

THE ERROR MESSAGE

The error message is the CV amplifier's reply to an invalid message. The error message can have two parts, the **error identifier** and the **returned message**.

- **ERROR IDENTIFIER**. The error identifier has two characters; the first is an exclamation mark (`"!"`), the second is an upper case alphabetic character to identify the type of error.
- **RETURNED MESSAGE.** Not all error types use a returned message. When present the returned message is a copy of the sent message, either upper or lower case depending on how far the message has been decoded before the error is detected.

ERRORS IN DETAIL

There are eight error types.

BUFFER OVERFLOW ERROR

This error is generated when a message has overflowed the CV amplifier's receive buffer. It is identified by the sequence (`"!B"`). It is used to indicate that the current message has been abandoned. It has no return message. All characters received by the CV amplifier are ignored until it receives a header character (`"<"`) to start a new message.

Example

`<B1,nnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnn nnnnnn<!B Message Buffer Full/>`

EXECUTION ERROR

This error is generated if the execution function for the command fails. It is identified by the sequence (`"!E"`).

Example

`<SY.RS,B38967/><!E SY.RS,B38967 Execution error/>`

NB 38967 is not a valid Baud rate.

INTERRUPTED ERROR

This error is generated when a header character ("<") is detected in the middle of a message. It is identified by the sequence ("!I"). It is used to indicate that the interrupted message has been abandoned and that the CV amplifier has started to receive a new message. An interruption is detected when the opening angle bracket of the new message is received; the Interrupted Error message is transmitted immediately.

Example

```
<B1,L<<!I Message Interrupted/>B2,L7/><b2,l1/>
```

NVM NOT READY ERROR

This error is generated if a control message is sent before the start up parameters have been read from Non Volatile Memory. It is identified by the sequence ("!N NVM not ready, please wait"). This error will clear shortly after power on; it is provided to acknowledge a message has been received but not acted upon.

OVERRUN ERROR

This error is generated when an unexpected end of message is detected whilst decoding. . It is identified by the sequence ("!A").

Example

```
<B1,/><!A B1./> Overrun error/>
```

PARSE ERROR

This error is generated when a destination or command field cannot be decoded as valid. It is identified by the sequence ("!P"). Its return message is an upper case copy of the sent message.

Example

```
<B1.MK,L8/><!P B1.MK,L8 Parse error/>
```
NB MK is not a valid sub destination.

TOKENISE ERROR

This error is generated when the message format is wrong. EG the destination field is too short or the command field is not separated by a comma. It is identified by the sequence ("!T"). Its return message is an upper case copy of the sent message. NB the return message may be omitted if the sent message was abnormally short.

Example

```
<B1R6,Q/><!T B1R6 Token error/>
```
NB no period between the main destination and the sub destination.

UNAVAILABLE ERROR

This error is generated when the destination channel or the command is not available for the block type. It is identified by the sequence ("!U"). Its return message is an upper case copy of the sent message.

Example

`<B1:LF,ID/><!U B1:LF,ID Unavailable error/>` NB. The block is not set as a crossover.

`<B9,LE/><!U B9,LE Unavailable error/>` NB. The block only contains Auxiliary out channels. The command is not available to Auxiliary out.

TABLE OF ERRORS

Error Type	ID	Fault Condition	Example Message	
			Faulty Message	Response
Buffer Full	B	Message too long.	`<GGGGGGGGGG...`	`<!B Message Buffer Full/>`
Execution	E	The execute function fails.	`<SY.RS,B300/>`	`<!E SY.RS,B300 Execution error/>`
Interrupted	I	Message interrupted before complete.	`<B1.L<`	`<!I Message Interrupted/>`
NVM	N	Waiting to finish reading Non Volatile Memory.	Any	`<("!N NVM not ready, please wait/>`
Overrun	A	Message incomplete.	`<B1,/>`	`<B1,/><!A B1./> Overrun error/>`
Parser	P	Error decoding message.	`<B1,L100/>`	`<!P B1,L100 Parse error/>`
Token	T	Error in message format.	`<B1L8/>`	`<!T B1L8 Token error/>`
Unavailable	U	Destination channel or sub destination not available to the block type.	`<B1:HF.PE,E/>`	`<!U B1:HF.PE,E Unavailable error/>`

EXAMPLES

Each example has a typical CV amplifier reply in red.

Auxiliary EQ

These commands are available to blocks that only contain auxiliary channels and are not set as a crossover.

```
<B5.AE,D/><b5.ae,d/>              (* Block 5 Auxiliary EQ disable.
<B5.AE,HP/><b5.ae,hp/>           (* Block 5 Auxiliary EQ set to High Pass.
<B5.AE,LP/><b5.ae,lp/>           (* Block 5 Auxiliary EQ set to Low Pass.
<B5.AE,F250/><b5.ae,f250/>       (* Block 5 Auxiliary EQ set to 250 Hz.
<B5.AE,BE18/><b5.ae,be18/>       (* Block 5 Auxiliary EQ slope Bessel 18dB per octave.
<B5.AE,BU12/><b5.ae,bu12/>       (* Block 5 Auxiliary EQ slope Butterworth 12dB per octave.
<B5.AE,LR24/><b5.ae,lr24/>       (* Block 5 Auxiliary EQ slope Linkwitz Riley 24dB per octave.
<B5.AE,Q/><b5.ae,lr24, f = 250, lp/>    (* Block 5 Auxiliary EQ query.
```

Auxiliary Level Slaving

These commands are available to blocks that only contain auxiliary channels.

```
<B6.AL,SE/><b6.al,se/>           (* Block 6 Auxiliary level slave enable.
<B6.AL,SD/><b6.al,sd/>           (* Block 6 Auxiliary level slave disable.
<B6.AL,SQ/><b6.al,sd/>           (* Block 6 Auxiliary level slave query.
```

Auxiliary Mute Slaving

These commands are available to blocks that only contain auxiliary channels.

```
<B6.AM,SE/><b6.am,se/>           (* Block 6 Auxiliary mute slave enable.
<B6.AM,SD/><b6.am,sd/>           (* Block 6 Auxiliary mute slave disable.
<B6.AM,SQ/><b6.am,sd/>           (* Block 6 Auxiliary mute query.
```

Auxiliary Source Slaving

These commands are available to blocks that only contain auxiliary channels.

```
<B6.AS,S2/><b6.as,s2/>           (* Block 6 Auxiliary source slave to block 2.
<B6.AS,SE/><b6.as,se/>           (* Block 6 Auxiliary source slave enable.
<B6.AS,SD/><b6.as,sd/>           (* Block 6 Auxiliary source slave disable.
<B6.AS,SQ/><b6.as,sd,2/>         (* Block 6 Auxiliary source slave query.
```

Baud

```
<SY.RS,B9600/><sy.rs,b=9600/>    (* Set Baud rate to 9600.
<SY.RS,BQ/><sy.rs,b=9600/>       (* Query Baud rate ).
```

Block Configure

```
<BL.CF,D/><bl.cf,d/>                              (* Block configure disable.
<BL.CF,E/><bl.cf,e/>                              (* Block configure enable.
<BL.CF,C=ST,ST,2WM,2WSM/><bl.cf,c= = st,st,2mo,2st,mo/>    (* Block configure
<BL.CF,Q/><bl.cf,q = st,st,2mo,2st,mo/>           (* Block configure query.
<B1.CF,Q/><b1.cf,q = st/>                         (* Block 1 configure query.
```

Boot Load Mode

```
<SY,BU1234/><sy,bu = ud/>        (* Unlock boot loader. NB PIN required.
<SY,BE/><sy,be = ue/>            (* Enable boot loader.
<SY,BR/><sy,br/>                 (* Boot loader system reset. Boot loader now active.
<SY,BL/><sy,bl = ld/>            (* Lock boot loader. NB boot loader is also disabled.
<SY,BD/><sy,bd = ld/>            (* Disable boot loader.
<SY,BQ/><sy,bq = ld/>            (* Query boot loader.
<SY,BR/><!E SY,BR Execution error/> (* System reset with boot loader disabled.
```

Crossover

```
<B3.XL,BU12/><b3.xl,bu12/>        (* Block 3 low crossover slope Butterworth 6dB per octave.
<B3.XH,BE18/><b3.xh,be18/>        (* Block 3 high crossover slope Bessel 18db per octave .
<B3.XL,LR24/><b3.xl,lr24/>        (* Block 3 low crossover slope Linkwitz Riley 24db per octave
<B4.XH,OE/><b4.xh,oe/>            (* Block 4 high crossover overlap enabled.
<B4.XL,OD/><b4.xl,od/>            (* Block 4 low crossover overlap disabled.
<B3.XH,F300/><b3.xh,f300/>        (* Block 3 high crossover slope set to 300Hz.
<B3.XH,Q/><b3.xh,lr24, f = 300, e, od/>    (* Block 3 low crossover query.
```

Crossover EQ Balance

```
<B4:HF.EQ,L6/><b4:hf.eq,l6/>      (* Block 4 high channel music EQ balance -6dB.
<B4:LF.EQ,LQ/><b4:lf.eq,l0/>      (* Block 4 low channel music EQ balance query.
```

Delay Build

```
<SY.DB,N/><sy.db,n/>              (* Start a new Delay Build.
<SY.DB,R/><sy.db,r/>              (* Reset, cancel the new Delay Build.
<B1.DB,M200/><b1.db,m200/>        (* Block 1 reserve 200ms in the Delay Build.
<B1.DB,MQ/><b1.db,m200/>          (* Block 1 query delay reserved in the Delay Build.
<SY.DB,AQ/><sy.db,a1082/>         (* Query delay available in the Delay Build.
<SY.DB,S/><sy.db,s/>              (* Set the Delay Build.
```

Delay

```
<B1.DY,MQ/><b1.dy,m200/>          (* Block 1 query assigned delay.
<B1.DY,MD20/><b1.dy,m180/>        (* Block 1 decrement assigned delay 20ms.
<B1.DY,MU5/><b1.dy,m185/>         (* Block 1 increment assigned delay 5ms.
<SY.DY,MQ/><sy.dy,m380/>          (* Query total assigned delay. NB Block 1 is stereo.
<SY.DY,R/><sy.dy,r/>              (* Reset delay.
```

GPIO

```
<SY.G1,MI/><sy.g1,mi/>            (* GPIO1 mode as Mute All Input.
<SY.G2,FO/><sy.g2,fo/>            (* GPIO2 mode as Fault Output.
<SY.G3,UI/><sy.g3,ui/>            (* GPIO3 mode as User Input.
<SY.G3,UO/><sy.g3,uo/>            (* GPIO3 mode as User Output.
<SY.G3,MQ/><sy.g3,m = uo/>        (* GPIO3 mode query.
<SY.G3,S/><sy.g3,s/>              (* GPIO3 set output state.
<SY.G3,C/><sy.g3,c/>              (* GPIO3 clear output state.
```

Initialisation

```
<SY,IF/><sy,if/>                  (* Set power up mode to factory.
<SY,IP/><sy,ip/>                  (* Set power up mode to last ('P' = previous ).
<SY,IQ/><sy,i = p/>               (* Query power up mode.
```

Invert

```
<B1,ID/><b1,id/>                  (* Block 1 output invert disable.
<B1,IE/><b1,ie/>                  (* Block 1 output invert enable.
<B4:HF,IE/><b4:hf,ie/>            (* Block 4 high channel output invert enable.
<B1,IQ/><b1,ie/>                  (* Block 1 output invert query.
```

Level

Set the level in a block.

`<B1,L9/><b1,l9/>`	(* Block 1 level to 9.
`<B1,LU7/><b1,l2/>`	(* Block 1 level up by 7.
`<B1,LD3/><b1,l5/>`	(* Block 1 level down by 3.
`<B1,LQ/><b1,l5,e/>`	(* Block 1 level query.
`<B1,LE/><b1,le/>`	(* Block 1 level parser control enable.
`<B1,LX/><b1,lx/>`	(* Block 1 level parser control disable.
`<B1,LQ/><b1,l5,x/>`	

Limiter

`<B1.LI,D/><b1.li,d/>`	(* Block 1 limiter disable.
`<B1.LI,E/><b1.li,e/>`	(* Block 1 limiter enable.
`<B1.LI,T24/><b1.li,t24/>`	(* Block 1 limiter threshold -24dB.
`<B1.LI,GQ/><b1.li,g=9dB/>`	(* Block 1 limiter gain query.
`<B1.LI,Q/><b1.li,t=24, e/>`	(* Block 1 limiter query.

Line Voltage

`<B1,V70/><b1,v70/>`	(* Block 1 line voltage to 70V.
`<B2,V100/><b2,v100/>`	(* Block 2 line voltage to 100V
`<B4,VE/><b4,ve/>`	(* Block 4 enable parser control of voltage.
`<B5,VX/><b5,vx/>`	(* Block 5 disable parser control of voltage.
`<B4,VQ/><b4,v70,e/>`	(* Block 4 voltage query.

Meters

`<C1.MI,Q/><c1.mi,q=-25dB/>`	(* Channel 1 input meter query.
`<C1.MO,Q/><c1.mo,q=-21dB/>`	(* Channel 1 output meter query.

Mute

`<B1,M/><b1,m/>`	(* Block 1 mute.
`<B1,O/><b1,o/>`	(* Block 1 un mute.
`<B1,Q/><b1,q = o/>`	(* Block 1 query mute status.

Name Labels

`<L1,N=CD/><l1,n = CD/>`	(* Set line 1's label to `CD`.
`<B3,N=Lounge/><b3,n = Lounge/>`	(* Set block 3's label to `Lounge`.
`<A4,N=Jazz/><a4,n = Jazz/>`	(* Set Auxiliary input 4's label to `Jazz`
`<B3,NQ/><b3,n = Lounge/>`	(* Query block 3's label.

Password

`<SY,K12345678/><sy,k12345678/>`	(* Set password. Old password = 1234,
	(* new password = 5678 .

Ping

`<SY,?/><sy,?/>`	(* Ping

Power

`<SY,WU/><sy,wu/>`	(* Wake from APD.
`<SY,PD/><sy,pd/>`	(* Power down.
`<SY,PU/><sy,pu/>`	(* Power up.
`<SY,PQ/><sy,pe/>`	(* Remote Power Down socket query.

Reset

`<SY,R/><sy,r/>` (* Reset to factory settings.

Room EQ

```
<B1.RE,D/><b1.re,d/>                    (* Block 1 room EQ disable.
<B1.RE,E/><b1.re,e/>                    (* Block 1 room EQ enable.
<B1.RE,Q/><b1.re,e/>                    (* Block 1 room EQ query enable state.
<B1.RL,F100/><b1.rl,f100/>             (* Block 1 room EQ low shelf  to 100Hz.
<B1.RH,F8000/><b1.rh,f8000/>           (* Block 1 room EQ high shelf  to 8000Hz.
<B1.RL,C5/><b1.rl,c5/>                  (* Block 1 room EQ low shelf  cut 5dB.
<B1.RH,B7/><b1.rh,b7/>                  (* Block 1 room EQ high shelf  boost 7dB.
<B1.RL,Q/><b1.rl,=100, -5, e/>         (* Block 1 room EQ low shelf  query.
<B1.RH,Q/><b1.rh,=8000, 7, e/>         (* Block 1 room EQ high shelf query.
<B1.RB,B6/><b1.rb,b6/>                  (* Block 1 room EQ bass boost 6dB.
<B1.RT,C5/><b1.rt,c5/>                  (* Block 1 room EQ treble cut 5dB.
<B1.RB,Q/><b1.rb,=6/>                   (* Block 1 room EQ bass query.
<B1.RT,Q/><b1.rt,=-5/>                  (* Block 1 room EQ treble query.
<B2.R2,F200/><b2.r2,f200/>             (* Block 2 room EQ band 2 to 200Hz.
<B2.R3,Q0.8/><b2.r3,q0.8/>             (* Block 3 room EQ band 3 Q to 0.8.
<B4.R7,B5/><b4.r7,b5/>                  (* Block 4 room EQ band 7 boost 5dB.
<B3.R4,C2/><b3.r4,c2/>                  (* Block 3 room EQ band 4 cut 2dB.
<B1.R5,Q/><b1.r5,=4000, 0.7, 0, e/>  (* Block 1 room EQ band 5 query.
```

RTC

```
<SY.RC,CD/><sy.rc,cd/>                  (* RTC calibration disable.
<SY.RC,CE/><sy.rc,ce/>                  (* RTC calibration enable.
<SY.RC,Q/><sy.rc,q0,0,e/>              (* RTC calibration query.
<SY.RC,CU5/><sy.rc,c5,e/>              (* RTC calibration plus 5.
<SY.RC,CD7/><sy.rc,c-2,e/>             (* RTC calibration minus 7.
<SY.RC,CS/><sy.rc,cs/>                  (* RTC calibration set.
```

Speaker Delay

```
<B3:LF.SD,U500/><b3:lf.sd,u500/>       (* Block  speaker delay set to 500us.
<B3:LF.SD,UQ/><b3:lf.sd,u500/>         (* Block  speaker delay query.
```

Speaker EQ

```
<B1.SE,D/><b1.se,d/>                    (* Block 1 speaker EQ disable.
<B1.SE,E/><b1.se,e/>                    (* Block 1 speaker EQ enable.
<B1.SE,Q/><b1.se,e/>                    (* Block 1 speaker EQ query enable state.
<B3:HF.S1,F100/><b3:hf.s1,f100/>       (* Block 3 high channel speaker EQ  band 1 100Hz.
<B1.S2,F400/><b1.s2,f400/>             (* Block 1 speaker EQ  band 2 400Hz.
<B2.S3,Q1.1/><b2.s3,q1.1/>             (* Block 2 speaker EQ  band 3 Q to 1.1.
<B3:LF.S5,C3/><b3:lf.s5,c3/>           (* Block 3 low channel speaker EQ  band 3 cut 3dB.
<B1.S5,B7/><b1.s5,b7/>                  (* Block 1 speaker EQ  band 5 boost 7dB.
<B1.S1,Q/><b1.s1,=60, 0.7, 0, e/>    (* Block 1 speaker EQ  query.
```

Source

Set the source of a block.

```
<B1,S3/><b1,s3/>                        (* Block 1 source to 3.
<B1,SA/><b1,sa/>                        (* Block 1 source from Aux in.
<B1,SL/><b1,sl/>                        (* Block 1 source from Line in.
<B1,SX/><b1,sx/>                        (* Block 1 source parser control disable.
<B1,SE/><b1,se/>                        (* Block 1 source parser control enable.
<B1,SS/><b1,ss/>                        (* Block 1 source stereo sum
<B1,SM/><b1,sm/>                        (* Block 1 source mono (stereo sum off ).
<B1,SQ/><b1,s1,0,l,e/>                  (* Block 1 source query.
```

Speaker Protect High pass filter

```
<B1.PF,PX/><b1.pf,px/>                    (* Block 1 protect filter parser control disable.
<B1.PF,PE/><b1.pf,pe/>                    (* Block 1 protect filter parser control enable.
<B4:HF.PF,D/><b4:hf.pf,d/>               (* Block 4 high channel protect filter disable.
<B4:HF.PF,E/><b4:hf.pf,e/>               (* Block 4 high channel protect filter enable.
<B1.PF,F80/><b1.pf,f80/>                  (* Block 1 protect filter set to 80Hz.
<B1.PF,Q/><b1.pf,=80,e,e/>               (* Block 1 protect filter query.
```

Text Field

```
<SY.TX,S=Some Text/><sy.tx,s=Some Text/>  (* Set text field to "Some Text"
<SY.TX,Q/><sy.tx,q = Label Text/>         (* Query text field.
```

Version

```
<SY.HV,Q/><sy.hv,q = V1.3/>               (* Hardware version for CV amplifier.
<SY.SV,Q/><sy.sv,q = V0.4/>               (* Software version for CV amplifier.
```